# Image Registration by Combining Thin-Plate Splines With a 3D Morphable Model

Vincent Gay-Bellile, Mathieu Perriollat, Adrien Bartoli, Patrick Sayd

▶ **To cite this version:**

Vincent Gay-Bellile, Mathieu Perriollat, Adrien Bartoli, Patrick Sayd. Image Registration by Combining Thin-Plate Splines With a 3D Morphable Model. International Conference on Image Processing, 2006, United States. hal-00094751

**HAL Id: hal-00094751**
**https://hal.science/hal-00094751**

Submitted on 14 Sep 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# IMAGE REGISTRATION BY COMBINING THIN-PLATE SPLINES WITH A 3D MORPHABLE MODEL

*Vincent Gay-Bellile* [1,2] *– Mathieu Perriollat* [1] *– Adrien Bartoli* [1] *– Patrick Sayd* [2]

[1] LASMEA – CNRS / Université Blaise Pascal – Clermont-Ferrand, France
[2] CEA Saclay, France.
Vincent.GAY-BELLILE@univ-bpclermont.fr

## ABSTRACT

Registering images of a deforming surface is a well-studied problem. It is common practice to describe the image deformation fields with Thin-Plate Splines. This has the advantage to involve small numbers of parameters, but has the drawback that the 3D surface is not explicitly reconstructed. We propose an image deformation model combining Thin-Plate Splines with 3D entities – a 3D control mesh and a camera – overcoming the above mentioned drawback. An original solution to the non-rigid image registration problem using this model is proposed and demonstrated on simulated and real data.

## 1. INTRODUCTION

Registering images of deformable surfaces is important for tasks such as video augmentation, dense Stucture-From-Motion, or deformation capture. This is a difficult problem since the appearance of imaged surfaces varies due to several phenomena such as camera pose, surface deformation, lighting and motion blur. Recovering a generic 3D surface, its deformations and the imaging sensor parameters from monocular video sequences is intrinsically ill-posed. For this reason, most work avoid a full 3D model by directly using image-based deformation models [1, 2, 3]. The obvious drawback of these approaches is that they do not reconstruct the 3D surface.

We propose a novel approach which jointly register the images and computes the 3D surface. It has two main originalities. First, we propose a mixed 3D and image-based generative model combining Thin-Plate Splines (TPS) with a 3D mesh and a camera. This model is dubbed 3D+TPS. It induces a piecewise smooth image deformation field while allowing one to reconstruct a 3D surface corresponding to each image of the sequence. In order to deal with the ill-posedness of the 3D surface and camera pose recovery, admissible surface deformations are learnt as a 3D Morphable Model [2, 4]. Second, we extend a tracking method that was successfully applied to twodimensional cases [1]. It consists in learning an interaction matrix, modeling as a Jacobian matrix does, the relationship between the image intensity variations and those of the model parameters.

Our 3D+TPS model is described in §2 and image registration in §3. Experimental results are reported in §4 and our conclusions are given in §5.

**Notation.** Vectors are typeset using bold fonts, *e.g.* $\mathbf{q}$, matrices using sans-serif fonts, *e.g.* $\mathsf{E}$, and scalars in italics, *e.g.* $\alpha$. Matrix and vector transposition in denoted as in $\mathsf{A}^\mathsf{T}$.

**Previous Work.** The registration of images of deformable objects using a single camera has received a growing attention over the past decade. Many approaches have been proposed, based on features or direct image intensity comparison.

Feature-based approaches locate image features on the model, then solve for the registration. For example, a highly efficient surface detection approach is proposed in [5]. The authors use a 2D regularized surface mesh in conjunction with a highly robust estimator to match feature points.

Direct approaches minimize an error expressed on image intensities. Active Appearance Models [1] are 2D learnt generative models that can be fitted to images to track deforming objects. They have been recently extended to 3D [6]. In [3], Radial Basis Mappings represent the transformation.

## 2. A GENERATIVE IMAGE MODEL

We present the image-based and 3D approaches to modeling image deformations, and then show how to combine them in a single model, drawing on the strengths of both approaches.

### 2.1. The Image-Based Part: Thin-Plate Splines

A TPS represents a smooth image deformation field. It maps a point $\mathbf{x}$ from the reference image $\mathcal{I}_0$ to the corresponding point $\mathbf{x}' = \tau(\mathbf{x}; \mathbf{q}_t) = \tau_t(\mathbf{x})$ in the target image $\mathcal{I}_t$, see *e.g.* [3] with:

$$\tau(\mathbf{x}; \mathbf{q}_t) = \mathsf{A}\mathbf{x} + \mathbf{y} + \sum_{j=1}^{m} \mathbf{w}_j \phi(\|\mathbf{x} - \mathbf{q}_{tj}\|),$$

where $(\mathsf{A}, \mathbf{y})$ represents a 2D affine transformation and the $\mathbf{w}_j$ and the $\mathbf{q}_{tj}$ are respectively the coefficients and the centers of

the transformation. The kernel function is $\phi(\mu) = \mu^2 \log(\mu)$. TPS are traditionally estimated from point correspondences, see *e.g.* [7].

### 2.2. The 3D Part: Control Mesh and Camera

A piecewise planar 3D control mesh approximating the surface is used along with a camera model to explain the deformations in the images. So as to deal with the ill-posedness inherent to deforming surface recovery, a 3D Morphable Model is used for the control mesh. The typical deformations are learnt prior to image registration, using PCA (Principal Component Analysis) on a collection of admissible meshes. More details are given in §4.1. The 3D mesh $\mathbf{Q}_t$ is thus expressed in terms of a mean mesh $\bar{\mathbf{E}}$ and $l$ eigenmeshes $\mathbf{E}_k$, and parameterized by view-dependent *configuration weights* $\boldsymbol{\alpha}_t$, so that a 3D vertex is given by:

$$\mathbf{Q}_{tj} = \bar{\mathbf{E}}_j + \sum_{k=1}^{l} \alpha_{tk} \mathbf{E}_{kj}.$$

The 3D mesh can be rotated and translated to account for camera pose: $\mathbf{Q}'_{tj} = \mathsf{R}(\mathbf{a}_t)\mathbf{Q}_{tj} + \mathbf{y}_t$, where $\mathbf{a}_t$ is a 3-vector containing the 3 rotation angles. A projective camera with fixed intrinsic parameters is used. Defining the vector of parameters $\mathbf{S}_t^\mathsf{T} = (\mathbf{a}_t^\mathsf{T}, \mathbf{y}_t^\mathsf{T}, \boldsymbol{\alpha}_t^\mathsf{T})$, the projection is written $\mathbf{q}_t = \Pi(\mathbf{S}_t)$, where $\mathbf{q}_t$ contains the vertices of the imaged mesh.

### 2.3. The 3D+TPS Model

The main idea for building the 3D+TPS model is that a TPS $\tau_t$ can be controlled by using as centers the projected vertices of the 3D control mesh in the reference and target image $t$. The registration error induced by the TPS will in turn constrain the 3D entities parameters. This tight coupling allows us to infer 3D information from images only.

The transfer function $\tau_t$ induced by the 3D+TPS model is used in two ways. First, in the interaction matrices learning stage, for generating images from locally perturbed model parameters. Second, in the registration stage, for warping the target image $\mathcal{I}_t$ onto the reference image. In the first case, $\tau_t^{-1}$ is required for warping the reference image onto the perturbed image, while in the second case, $\tau_t$ is required. We propose an efficient approximation of $\tau^{-1}$ for image warping in §3.2.

To sum up, our 3D+TPS model has the advantage of explicitly involving a simple 3D surface mesh and camera pose and produces a smooth deformation field.

## 3. REGISTERING IMAGES

### 3.1. The Error Function

A great variety of feature and intensity based error function are proposed in the litterature. We adopt the direct approach and minimize the sum of squares of intensity differences over pixels $\mathcal{K}$ which are Canny edge points in the reference image:

$$\mathcal{C}(\mathbf{S}_t) = \sum_{\mathbf{x} \in \mathcal{K}} \left( \mathcal{I}_0[\mathbf{x}] - \mathcal{I}_t\left[\tau(\mathbf{x}; \Pi(\mathbf{S}_t))\right] \right)^2. \quad (1)$$

We work with edge points only because they carry the most important texture information. Minimizing $\mathcal{C}$ is a nonlinear least squares problem. One of the most convincing approaches in the litterature is [8]. The error criterion (1) is linearized around $\mathbf{S}_0$ yielding:

$$\delta\mathbf{S}_0 \quad = \mathcal{F} \cdot \text{vect}(\delta\mathcal{I}), \quad (2)$$

where vect is the matrix vectorization operator. A closed-formed expression is derived for matrix $\mathcal{F}$ using the inverse Jacobian image. Cootes *et al.* [1] propose to estimate $\mathcal{F}$ from training images obtained by perturbing the generative model parameters around $\mathbf{S}_0$. The learnt matrix $\mathcal{F}$ is called the *interaction matrix*.

### 3.2. Learning the Interaction Matrix

Perturbations are drawn randomly and selected if the maximum displacement of image vertices is below some threshold $\gamma$ that we typically choose as a few pixels. For each selected perturbation, a synthetic image is generated in order to compute the change in appearance, see figure 1. Given the generative model parameters $\mathbf{S}^i$ for the perturbed image, we form the transfer function $\tau^{-1}$ and warp the texture image:

$$\tilde{\mathcal{I}}^i[\mathbf{x}] = \mathcal{I}_0[\tau^{-1}(\mathbf{x}; \Pi(\mathbf{S}^i))].$$

This function is implemented using a TPS interpolating the vertices. To compute the TPS coefficients, we generate a regular grid where vertices play the role of centers for the TPS. Thanks to the vertex correspondences between the reference and the perturbed images, we lineary compute the TPS parameters and so the transfer function $\tau^{-1}$.

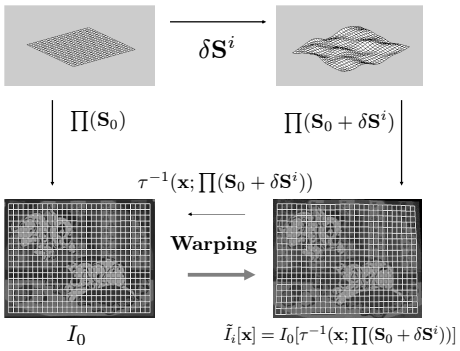


**Fig. 1**. Training images, needed to learn the interaction matrices, are generated by perturbing the model parameters.

### 3.3. Registration of an Image Sequence

We proceed as follows. We initialize $\mathbf{S}_t$ to any initial guess, for example $\mathbf{S}_t \leftarrow \mathbf{S}_{t-1}$. We warp the current image $\mathcal{I}_t$ to $\tilde{\mathcal{I}}_t$ using the mapping induced by $\mathbf{S}_t$, and compute the difference image $\delta\mathcal{I}_t = \mathcal{I}_0 - \tilde{\mathcal{I}}_t$. The local update $\delta\mathbf{S}_0$ is then computed from (2) as $\delta\mathbf{S}_0 = \mathcal{F} \cdot \text{vect}(\delta\mathcal{I}_t)$. It must be composed with the current $\mathbf{S}_t$ in order to update it, as illustrated on figure 2. This is a forward compositional strategy [9]. How to compose the local mapping correction $\delta\mathbf{S}_0$ with $\mathbf{S}_t$ is not straightforward. We solve this problem by mapping the control mesh vertices from the reference to the target view giving $\mathbf{q}_t = \tau(\Pi(\mathbf{S}_0 + \delta\mathbf{S}_0); \mathbf{S}_t)$, and minimizing the discrepancy between these vertices and the vertices predicted by the model, *i.e.* the reprojection error, over $\mathbf{S}_t$:

$$\min_{\mathbf{S}_t} \|\mathbf{q}_t - \Pi(\mathbf{S}_t)\|^2. \tag{3}$$

The minimization is solved using the nonlinear least squares algorithm Levenberg-Marquardt.

As underlined above, the linear relationship (2) represents a local approximation of the cost function around $\mathbf{S}_0$. Obviously, the validity of the approximation is conditioned upon the magnitude $\gamma$ (expressed in pixels), of the perturbation used for generating the training images. In order to increase the speed of convergence and widen the size of the bassin of convergence, we learn not only one, but rather a serie $\mathcal{F}_1, \ldots, \mathcal{F}_\kappa$ of interaction matrices, with a gradually lower perturbation magnitude. This forms a coarse to fine set of linear approximations to the error function, that we apply in turn.

This approach is different from the one in [6], which penalizes a 2D Active Appearance Model, by jointly computing a 3D Morphable Model. In the approach we propose, the 3D model and the TPS are represented with the same set of parameters. One of the differences with [1] is that they assumed that the domain where the linear relationship (2) is valid covers the whole set of registrations, thus avoiding the need of the difficult composition step. This however not appears to be a valid choice in practice.

## 4. EXPERIMENTAL RESULTS

### 4.1. The Control Mesh

Depending on the kind of surface that one may want to register, different surface generation schemes are used. For instance, the 3D Morphable Model proposed in [4] can be used for faces. We are interested in registering images of surfaces such as a rug or a sheet of paper, and follow [2] to generate a set of training meshes by deforming a regular, flat mesh by randomly changing the angles between the different facets.

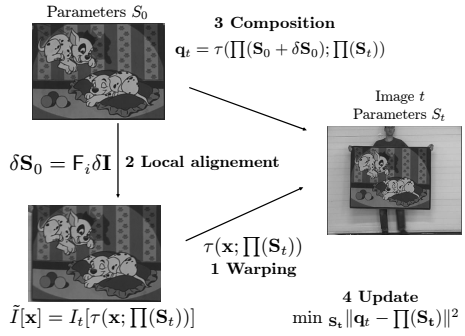


**Fig. 2**. Our image registration algorithm follows the forward compositionnal strategy, see text for details.

### 4.2. Simulated Data

In order to assess the behaviour of our algorithm in different conditions, we synthesized images under controlled conditions. Given a reference image, we applied a random perturbation to our model such that the mean rigid displacement of the pixels, caused by the relative displacement between the camera and the control mesh, is $\delta_R$, and the mean non-rigid displacement of the pixels, caused by the deformation of the control mesh, is $\delta_{NR}$. We added gaussian noise, with variance $\sigma$ % of the maximum greylevel value, to the warped image. We varied each of these parameters independently, using the following default values: $\delta_R = 5$ pixels, $\delta_{NR} = 3$ pixels, $\sigma = 1\%$ while measuring the residual error defined as the mean of Euclidean distance between the vertices of the mesh which generated the warped image, and those of the estimated mesh. Figure 3 shows the results we obtained. We observed on figure 3 (a) that when the magnitude of the perturbation is greater than 20 pixels, the registration efficiency quickly decreases. Those perturbation magnitudes have actually not been learnt, causing the linear approximation being less accurate. We expect the average displacement between consecutive images to be far less than 20 pixels in real cases. Figure 3 (b) shows that the alignment error in pixels is approximatively linear in the variance of the noise on image intensities. In practice, one can expect the noise magnitude to be in the order of 2% of the maximum grey value, making our algorithm well-adapted to many real image sequences.

### 4.3. Real Data

We tested our algorithm on several image sequences. One of them, consisting of 40 images see figure 5, is used to demonstrate our approach. To initialize the tracker, we made the assumption that the 3D mesh associated to the first image was flat. Consequently, the initialisation problem is equivalent to estimating the relative pose of a plane. Even if some images

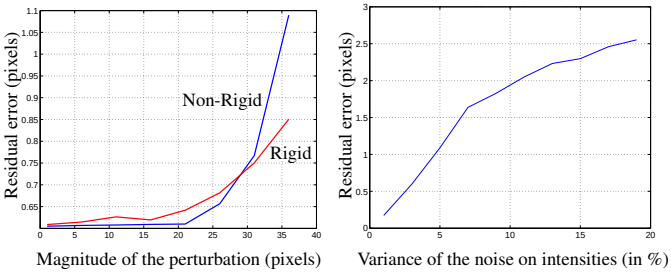


**Fig. 3**. Registration of simulated data. The results are median over 100 trials. (a) shows the residual error when the magnitude of the perturbation $\delta_N$ or $\delta_{NR}$ is varied. (b) shows the residual error for varying noise on the image intensities.

of the sequence are blurred, our algorithm achieved successful alignment. In some cases, however, the model drifted away from its ideal position due to lack of constraints in the texture, making some contour points sliding along their edge. Figure 4 shows the composition error and the registration error for each frame. We observed that the composition error, the one minimized in equation (3), is kept around a pixel, meaning that the composition step is successful. The registration error, proportional to equation (1), and expressed in image intensity unit, is kept around typical values, indicating that our model reliably fits the images. The algorithm has been implemented in Matlab, the registation is done at about 7s per image on a pentium IV 3GHz.

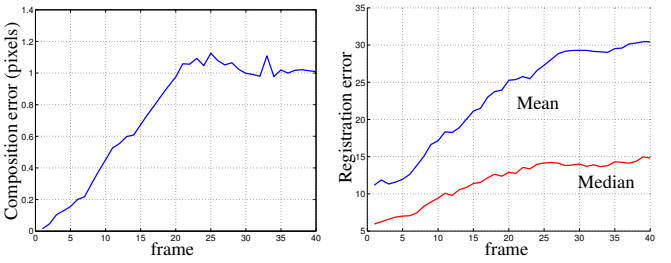


**Fig. 4**. The composition (a) and the registration (b) errors for the sequence shown in figure (5)

## 5. CONCLUSIONS

We proposed a novel generic approach to image registration based on a mixed 3D and image-based model. Combining TPS with a 3D mesh and a camera yields smooth image deformation fields while allowing one to recover a 3D surface for each image of a sequence. We plan to extend the method to deal with occlusions, by exploiting the reconstructed 3D surface to predict self-occlusions.

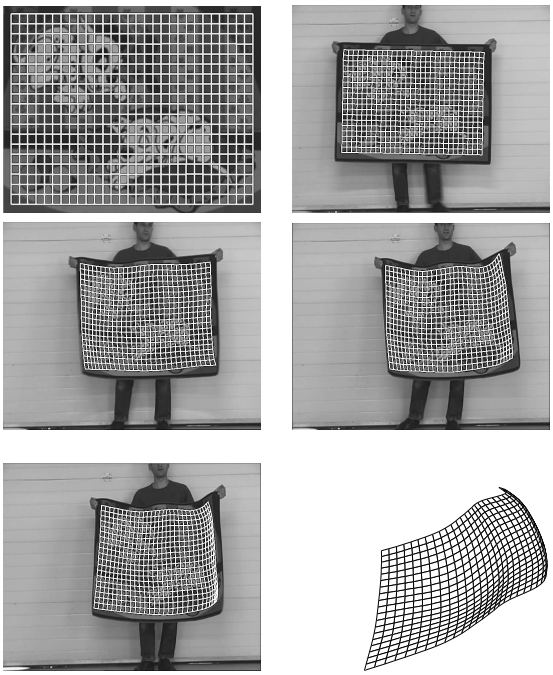

**Fig. 5**. (Top left) the reference image and his associated mesh. (Next) registration of an image sequence, the projected 3D mesh is shown in white. (Bottom right) the recovered 3D mesh for the last image.

## 6. REFERENCES


[1] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," PAMI, 23(6):681-685, 2001.

[2] M. Salzmann, S. Ilic, and P. Fua, "Physically valid shape parameterization for monocular 3-D deformable surface tracking," in BMVC, 2005.

[3] A. Bartoli and A. Zisserman, "Direct estimation of non-rigid registrations," in BMVC, 2004.

[4] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," PAMI, 25(9), September 2003.

[5] J. Pilet, V. Lepetit, and P. Fua, "Real-time non-rigid surface detection," in CVPR, 2005.

[6] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2D+3D active appearance models," in CVPR, 2004.

[7] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," PAMI, 11(6):567-585, June 1989.

[8] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," PAMI, 20(10):1025-1039, 1998.

[9] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," IJCV, 56(3):221-255, February 2004.